A Novel DCT domain CRT-based watermarking scheme for image authentication surviving JPEG compression

Phua, Jiliang E.; Patra, Jagdish Chandra; Bornand, Cedric

2010

Patra, J. C., Phua, J. E., & Bornand, C. (2010). A novel DCT domain CRT-based watermarking scheme for image authentication surviving JPEG compression. Digital Signal Processing, 20(6), 1597-1611.

https://hdl.handle.net/10356/94340

https://doi.org/10.1016/j.dsp.2010.03.010

A novel DCT domain CRT-based watermarking scheme for image authentication surviving JPEG compression

Jagdish C. Patra [a,*], Jiliang E. Phua [b], Cedric Bornand [c]

[a] School of Computer Engineering, Nanyang Technological University, Singapore
[b] Institute for Infocomm Research, Singapore
[c] University of Applied Sciences, HEIG-VD, Yverdon-les-Bains, Switzerland

ABSTRACT

Digital watermarking techniques have been proposed for copyright protection and authentication of multimedia data. In this paper, we propose a novel Chinese Remainder Theorem (CRT)-based technique for digital watermarking in the Discrete Cosine Transform (DCT) domain that is robust to several common attacks. We compared the performance of the proposed technique with recently proposed Singular Value Decomposition (SVD)-based and spatial CRT-based watermarking schemes. Experimental results have shown that the proposed technique successfully makes the watermark perceptually invisible and has better robustness to common image manipulation techniques such as JPEG compression, brightening and sharpening effects compared to the spatial domain-based CRT scheme. The proposed scheme is able to achieve a Tamper Assessment Function (TAF) value of less than 10% when the watermarked image undergoes JPEG compression between a range of 50 to 70%, where as, the spatial CRT-based scheme produce TAF value of more than 35% and the SVD-based scheme produces TAF value between 10 to 40% depending on the host image, for the same range of compression. When the watermark capacity is doubled, the proposed technique is still able to maintain imperceptibility and low TAF values, for most of the attacks.

1. Introduction

In recent years, image authentication has become a great challenge, mainly as a result of the rapid growth in computing power, Internet technology, electronic commerce, multimedia market, medical imaging applications, and the increasing interest in the Web applications. The sheer amount of information on the web is overwhelming — there are millions of text, image, video and animation files. Thus a huge amount of multimedia data is easily accessible for everyone through the Internet. Multimedia data in digital format can be modified or tampered with ease using a variety of available image processing tools, whether it is malicious or not. In other words, we cannot be sure if the image we have received from the Internet is authentic. The protection of intellectual property rights is another increasingly important issue with a large number of digital images interchanged on the Internet everyday. So, it is a challenging problem to ensure the integrity of received images as well as the original ownership for potential security loopholes of the public Internet.

Most of these concerns can be effectively addressed by watermarking techniques, owing to their three important characteristics; imperceptibility, inseparability from the cover content, and its inherent ability to undergo the same transformations as experienced by the cover content. In addition, it can be employed on different forms of digital media, such as, text, audio, image, graphic, movie and 3D models. Digital watermarking consists of embedding data at the content-level of digital me-

---

* Corresponding author.
E-mail addresses: aspatra@ntu.edu.sg (J.C. Patra), jephua@i2r.a-star.edu.sg (J.E. Phua), cedric.bornand@heig-vd.ch (C. Bornand).

dia under constraints on imperceptibility, security and robustness to attacks. In contrast to cryptography, which immediately arouses suspicion of something secret or valuable, the watermarking technique hides a watermark (also called digital signature) within the digital host without any noticeable change. The digital signature embedded remains invisible and cannot be removed easily under certain manipulations, e.g., compression and tampering operations. Only the authorized recipient of the digital content can extract the watermark from the watermarked content with the knowledge of some key information. As such, content integrity and intellectual protection can be provided.

For an effective watermarking technique [1–3], the watermark should be: (i) imperceptible: there should not be any visible difference between the watermarked content and the original, (ii) secure: no other watermark other than the embedded watermark should be extracted from the watermarked content. This reduces ambiguity on the ownership of the content. No one without the secret key may know the locations of where the watermark is embedded in the host, (iii) robust: the watermark should be able to withstand to some extent of content manipulation. There is no such watermark scheme that can perform well under all hostile attacks. However, with the growing need of sophisticated watermarking applications, we need a scheme that should perform well under a specific set of conceivable attacks.

Digital image watermarking techniques proposed so far can be divided into two main groups: those which embed watermark directly in the spatial domain [4] and those operating in a transformed domain, e.g., the frequency domain [2]. Techniques can also be distinguished according to the way the watermark is extracted from the possibly distorted version of the marked image. There are three main techniques: (i) blind: this scheme only requires a key for extraction of the watermark, (ii) semi-blind: the original watermark is required for extraction, (iii) non-blind: the original watermark and host image are required for extraction.

The research in watermarking techniques covers a wide area. Many different techniques have been devised, e.g., cryptography and quantization-based embedding to improve security [5,6], the discrete cosine transform (DCT) to make use of properties in the frequency domain [7–9] and using human visual system based scheme to make use of the human's perceptivity of images [10]. A wavelet transform-based [11] and an independent component analysis-based [12] blind watermarking scheme with interesting results have been proposed. Some of the other transform-domain watermarking schemes have been proposed in [13–15]. Recently there have been a series of papers on SVD-based watermarking scheme [16–22]. Many published schemes rely on similar principles but they differ in their implementation methodology and provide certain advantage in their domain of implementation.

Watermarking algorithms can be implemented either in software or hardware. In a *software* implementation, the algorithm is executed by a set of codes running on a microprocessor or on an embedded processor. Some of the advantages of software implementation are as follows. The designer needs to focus only on high level implementation of the algorithm instead of worrying about the hardware elements such as flip-flops, RAMS, and logic gates. Besides, the designer has the wealth of software libraries to realize various data operations. On the other hand, in hardware based implementation, all operations of the watermarking algorithm are carried out in custom-built circuitry, such as VLSI chips. Hardware implementation is essential for low power, real-time performance, high reliability, low cost applications, and also for easy integration with existing consumer electronic devices. For example, watermarking chips can be integrated with any existing digital still image camera. The hardware modules can also be integrated with a JPEG codec. Several hardware implementation issues and platforms for different watermarking algorithms for images and videos have been reported in [23].

The Chinese remainder theorem (CRT) [24] has been used in several engineering applications, such as, hierarchical access control by assigning cryptographic keys [25], secret sharing of stego-images [26], construction of quasi-cyclic codes [27], residue number systems [28] and oblivious data transfer mechanism [29]. Based on the CRT properties, a CRT-based spatial watermarking scheme with preliminary results has been reported in [30]. Detailed study on the spatial domain CRT-based watermarking scheme for multimedia content authentication has been recently reported in [31]. Although this technique is computationally fast and robust to several common attacks, its main drawback is its inability to withstand JPEG compression. The main purpose of using CRT in watermarking is its added security. For example, by selecting a set of relatively prime numbers $\mu = \{M_1, M_2, \ldots, M_r\}$ and using CRT, a large integer Z can be represented by a set of smaller integers $\{R_1, R_2, \ldots, R_r\}$. In comparison to SVD, the CRT-based watermarking scheme provides several advantages. Firstly, it is very difficult to get back the original integer Z without the knowledge of μ. This fact provides additional security in the proposed scheme. Moreover, since CRT is based on simultaneous congruence and modular arithmetic, it is computationally more efficient than computation of SVD.

In this paper we propose a novel watermarking technique for image authentication that utilizes CRT in DCT domain. Several reversible transform domains like discrete wavelet transform (DWT) and discrete Fourier transform (DFT) have been used in watermarking. We have chosen DCT because of its lower computational complexity and because of the fact that DCT has been used in the JPEG compression algorithm. We have shown that the proposed DCT-domain technique can withstand image manipulations quite well. Some of the major deficiencies of the recently proposed spatial domain CRT-based watermarking scheme [31] are as follows: (i) this scheme cannot withstand brightening and sharpening attacks and (ii) it cannot withstand any JPEG compression, a common image manipulation tool used in almost all digital imaging devices.

Our proposed scheme attempts to overcome the problems faced in [31]. In the proposed scheme the host image is first divided into small blocks and DCT is performed on the block. The watermark bit is embedded either on the DC or on low frequency AC component of the DCT by using CRT. We have compared performance of the proposed scheme with a recently reported SVD-based scheme [21] and a CRT-based spatial scheme [31], and shown its superior performance by conducting different attacks to the watermarked image. Especially the proposed technique is able to withstand JPEG compression to a great extent. The proposed scheme is able to achieve high peak signal to noise ratio (PSNR) and low tamper assessment function (TAF) values in case of most of the attacks. We have considered several attacks, such as, cropping, tampering, addition of noise, brightening, sharpening and JPEG compression. In the proposed scheme, the TAF value remains below 5% for cropping attack which is comparable to that of [31], but much lower than that of [21]. The proposed scheme maintains a low TAF value (below 10%) when the watermarked image undergoes JPEG compression with range of 50–90%, whereas, the other two schemes produce much higher TAF value, thus making the extracted watermark unrecognizable. In addition, the proposed scheme maintains its superiority in terms of TAF and PSNR over the other two schemes when the embedded watermark capacity is doubled. However, some of the other attacks, e.g., scaling, rotation, filtering and geometric attacks [34], are not considered in this paper.

The rest of the paper is as follows: The mathematics of CRT is introduced in Section 2. We briefly introduce an existing SVD-based scheme and spatial CRT-based scheme in Section 3. Next, the proposed CRT-based DCT watermarking scheme is explained in Section 4. The experimental results and performance comparison with the two schemes given in Section 3 have been provided in Section 5. Finally, conclusions and discussions on this study have been made in Section 6.

2. Chinese remainder theorem

2.1. The CRT

The CRT can be compactly stated as follows. Let μ be a set of r integers given by $\mu = \{M_1, M_2, \ldots, M_r\}$, such that any two M_i are pair-wise relatively prime. Let a set of r simultaneous congruences be given by [32]

$$Z \equiv R_i \pmod{M_i}, \tag{1}$$

where R_i, $i = 1, 2, \ldots, r$, are called residues. The solution for the integer Z can be found as

$$Z \equiv \left(\sum_{i=1}^{r} R_i \frac{M}{M_i} K_i \right) \pmod{M}, \tag{2}$$

where $M = M_1 \cdot M_2 \cdot \cdots \cdot M_r$ and K_i are determined from

$$K_i \frac{M}{M_i} \equiv 1 \pmod{M_i}. \tag{3}$$

Let us take a simple example with $r = 2$ to illustrate CRT. Let $M_1 = 6$, $M_2 = 11$. Let the two congruences be given as $Z \equiv 4 \pmod 6$ and $Z \equiv 8 \pmod{11}$. Thus, $R_1 = 4$, $R_2 = 8$. In order to find the value of Z, compute $M = M_1 \cdot M_2 = 66$. K_1 and K_2 are to be determined such that (3) is satisfied. That is, $(K_1 \frac{66}{6}) \equiv 1 \pmod 6$ and $(K_2 \frac{66}{11}) \equiv 1 \pmod{11}$. We can see that for $K_1 = 5$ and $K_2 = 2$, these two congruences are satisfied. Now Z is determined as $Z \equiv (4 \cdot \frac{66}{6} \cdot 5 + 8 \cdot \frac{66}{11} \cdot 2) \pmod{66} = 52$.

2.2. The inverse CRT

Given $\mu = \{M_1, M_2, \ldots, M_r\}$ and $M = M_1 \cdot M_2 \cdot \cdots \cdot M_r$, the objective of the inverse CRT is to represent any integer $Z, \{0 < Z \leqslant M - 1\}$ by a set of integers $Z = \{R_1, R_2, \ldots, R_r\}$. The R_i, $i = 1, 2, \ldots, r$, are obtained from the following congruences:

$$Z \equiv R_i \pmod{M_i}. \tag{4}$$

Let us take the previous example in which $M_1 = 6$ and $M_2 = 11$. Therefore, $M = M_1 \cdot M_2 = 66$. Let the given integer be $Z = 52$. Using (4), $52 \equiv R_1 \pmod 6$ and $52 \equiv R_2 \pmod{11}$. Thus, we get $R_1 = 4$ and $R_2 = 8$. Therefore, Z can be represented as $Z = \{4, 8\}$. For detailed discussions on CRT, one can refer to any textbook on number theory or cryptography [32,33].

2.3. Proposed CRT application

From the previous example, with $r = 2$, the absolute difference between R_1 and R_2 can be represented by d, which is given by

$$d = |R_1 - R_2|. \tag{5}$$

The maximum value of d can be found by taking the larger of the two moduli, M_1 and M_2 and subtracting one from it. It is represented by D as:

$$D = \max\{M_1, M_2\} - 1. \tag{6}$$

3. Existing watermarking schemes

In this section, we briefly discuss two different watermarking schemes and their weaknesses. Chang et al. [21] proposed an elegant SVD-based watermarking scheme which is represented as Scheme 1. Patra et al. [31] proposed a CRT-based watermarking scheme in the spatial domain which is represented as Scheme 2.

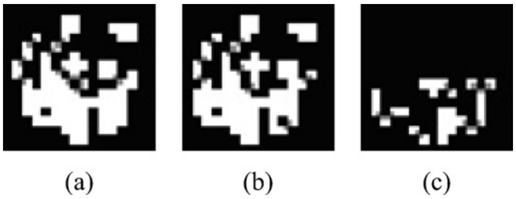


Fig. 1. Watermarking Scheme 1 [21]: (a) original 16 × 16 watermark, (b) extracted 16 × 16 watermark, (c) difference between original and extracted watermarks.

3.1. Watermarking Scheme 1

Let A be a matrix of size M x N representing a grayscale image. The elements of A will have values between 0 and 255, for 8-bit representation of pixels. Using singular value decomposition the matrix A can be decomposed into three matrices as follows:

$$\begin{aligned} A &= UDV^T \\ &= \begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1M} \\ u_{21} & u_{22} & \cdots & u_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ u_{M1} & u_{M2} & \cdots & u_{MM} \end{bmatrix} \begin{bmatrix} \lambda_1 & 0 & 0 & \cdots & 0 \\ 0 & \lambda_2 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \lambda_M \end{bmatrix} \begin{bmatrix} v_{11} & v_{21} & \cdots & v_{N1} \\ v_{12} & v_{22} & \cdots & v_{N2} \\ \vdots & \vdots & \ddots & \vdots \\ v_{1N} & v_{2N} & \cdots & v_{NN} \end{bmatrix}, \end{aligned} \tag{7}$$

where $U(M \times M)$ and $V(N \times N)$ represent the orthogonal matrices of A, and $D(M \times N)$ (same dimension as A) represents a diagonal matrix. The eigen vectors of AA^T make up the columns of U and the eigen vectors of A^TA are the columns of V. The diagonal elements of D represent square root of eigen values of either A^TA or AA^T, and these are arranged in descending order, i.e., $\lambda_1 > \lambda_2 > \lambda_3 > \cdots > \lambda_M$. Rank of A is defined as the number of non-zero diagonal elements of D.

The SVD-based watermarking scheme proposed by Chang et al. [21] uses a block-based SVD technique. The host image is a grayscale image and the watermark is a binary image. First, the host image is divided into blocks of $n \times n$ pixels. A single watermark bit is embedded in a block. The blocks are chosen randomly using a pseudo-random number generator (PRNG) based on their rank. The blocks with higher ranks are selected first before those of lower ranks.

In order to embed a watermark bit in a block the following procedure is adopted. We first perform a SVD transformation on the selected block. Let c_1 and c_2 denote the second and third elements (u_{21} and u_{31}) of the first column of U matrix, respectively. The embedding rules are as follows:

To embed a watermark bit '1', the value of $(c_1 - c_2)$ should be positive and its magnitude is greater than a strength factor, β. If this condition is not satisfied, c_1 and c_2 are modified as c_1' and c_2', respectively, such that this condition is satisfied. The modification rule is as follows:

$$c_1' = b + \beta/2, \qquad c_2' = b - \beta/2, \tag{8}$$

where $b = (|c_1| + |c_2|)/2$.

To embed a watermark bit '0', the value of $(c_1 - c_2)$ should be negative and its magnitude is greater than the strength factor, β. If this condition is not satisfied, c_1 and c_2 are modified as c_1' and c_2', respectively, such that this condition is satisfied. The modification rule is as follows:

$$c_1' = b - \beta/2, \qquad c_2' = b + \beta/2. \tag{9}$$

To construct the watermarked block, an inverse SVD transformation is performed on the modified U matrix with the original D and V matrices. This watermarked block then replaces the original selected block in the host image. This embedding process is repeated until all the watermark bits have been embedded.

The initial steps of the extraction procedure are similar to that of the embedding process up to and including the selection of the coefficients c_1 and c_2. The value of $(c_1 - c_2)$ determines the value of the extracted watermark bit. A positive difference indicates that the watermark bit is a '1', whereas a negative difference would imply that a watermark bit '0' is extracted.

The scheme is quite promising. However, the main weakness of this scheme is its reliance on the use of rank in selecting an image block for embedding watermark bits. This reliance causes the scheme to be less resistant to attacks, which alters the rank of the image blocks. The selection of image blocks based on higher ranks would lower the distortion level of the watermarked image. However, the rank is not a reliable feature as it is not stable. The rank of the selected block could change after modifications are made to the elements c_1 and c_2. Therefore, without any tampering to the watermarked image, the extracted watermark could be corrupted due to a change in rank of the block. Fig. 1 shows the corruption of the extracted watermark due to changes in the rank. In general, when the strength factor increases, the likelihood of change in the rank of a block increases. This leads to a higher level of corruption in the extracted watermark and distortion in the watermarked image.

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
X							
Y		Z					

Fig. 2. Watermarking Scheme 2: bit representation of a pixel.

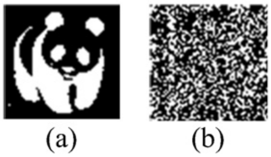

(a) (b)

Fig. 3. Watermarking Scheme 2 [31]: (a) original watermark, (b) extracted watermark after 90% JPEG compression.

3.2. Watermarking Scheme 2

The CRT-based watermarking scheme proposed by Patra et al. [31] uses the CRT methodology in the spatial domain. Let A be a matrix of size $M \times N$ representing a grayscale image. The elements of A will have values between 0 and 255, for 8-bit representation of the pixels. Each pixel is represented as shown in Fig. 2.

The host image is a grayscale image and the watermark is a binary image. First, the host image is divided into blocks of $n \times n$ pixels. A single watermark bit is embedded in a block. A random pixel, X is then selected from the block using a random number generator and the six least significant bits of X (Fig. 2, bits 0–5) are converted into a decimal value Z. The two most significant bits (Fig. 2, bits 6–7) are then converted into decimal value Y as shown in Fig. 2. Inverse CRT is then applied to Z to get the values of R_1 and R_2. The embedding rules are as follows.

To embed a watermark bit ‘1’, the following condition should satisfy: $R_1 \geqslant R_2$. If this condition is not satisfied, R_1 and R_2 are modified by adding or subtracting one from Z to obtain the modified Z'. To embed a watermark bit ‘0’, the following condition should satisfy: $R_1 < R_2$. If this condition is not satisfied, R_1 and R_2 are modified by adding or subtracting one from Z to obtain the modified Z'. Then Z' is combined with Y to get the new watermarked pixel X'. To construct the watermarked block, X' is used to create the watermarked image. This watermarked block then replaces the original selected block in the host image. This embedding process is repeated until all the watermark bits have been embedded.

The initial steps of the extraction procedure are similar to that of the embedding process up to and including the computation of the coefficients R_1 and R_2. The values of R_1 and R_2 determines the value of the extracted watermark bit. If $R_1 \geqslant R_2$ then it implies that the watermark bit is a ‘1’, otherwise the watermark bit is a ‘0’.

The scheme is very fast and computationally efficient. Also, the watermarked image maintains a high visual quality. However, its main weakness is that this scheme chooses the spatial domain for embedding the watermark bits. Spatial domain methods are usually not preferred due to the fact that they are not robust to common image processing applications especially to lossy compression such as the common JPEG compression. As such, the watermarked information contained in the pixel value even if slightly modified would result in an erroneous extracted watermark. This is shown in the Fig. 3 where the watermarked image undergoes a 90% JPEG compression and the watermark is extracted. Fig. 3b shows that even for a slight compression, the scheme fails to extract the watermark.

4. Proposed CRT-based DCT watermarking scheme

The embedding and extraction procedure of the proposed scheme is based on CRT technique in DCT domain. This scheme attempts to provide better security through the use of CRT, better robustness against attacks and needs minimal knowledge during the extraction phase. We will cover some guidelines for developing this scheme. The design of the CRT algorithm used and the choice of embedding locations in the DCT domain will be mentioned. Finally, the details of the embedding and extraction process will be presented.

4.1. DCT domain

One of the major weaknesses of Scheme 2 is that it fails for JPEG image compression. It was observed that it is due to the spatial domain in which the watermark is processed. In order to improve the robustness against JPEG compression, the proposed scheme operates in the DCT domain, the same domain in which JPEG compression algorithm is based on.

For the proposed scheme, we would be using a block based approach. With reference to Fig. 4, the initial step is to divide the host image into blocks of 8×8 pixels. This is also the block size used in the JPEG compression algorithm. The blocks are then converted into the DCT domain where embedding of watermark information will be processed. After the embedding process, the watermarked DCT blocks will then undergo inverse DCT to construct the watermarked image.

For extraction of watermark, the watermarked image is first divided into blocks of 8×8 pixels. These blocks then undergo DCT conversion and the watermark is extracted as shown in Fig. 5. As such, the proposed watermarking scheme does the processing in the DCT domain instead of the spatial domain as in Scheme 2.

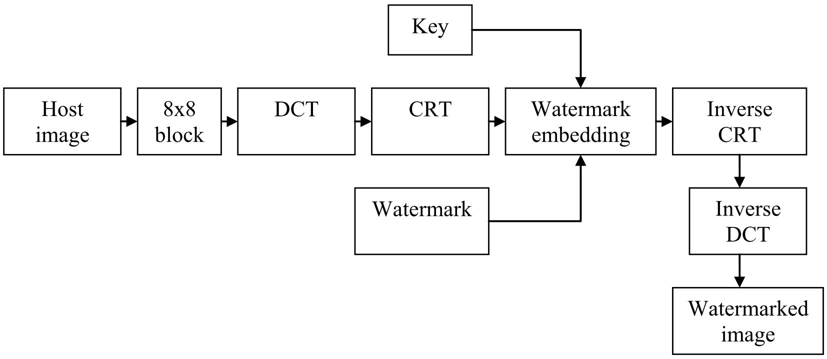


Fig. 4. Proposed scheme: overview of watermark embedding process.

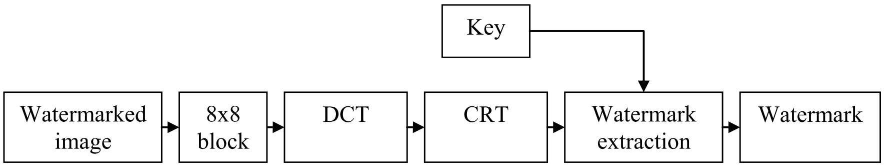


Fig. 5. Proposed scheme: overview of watermark extraction process.

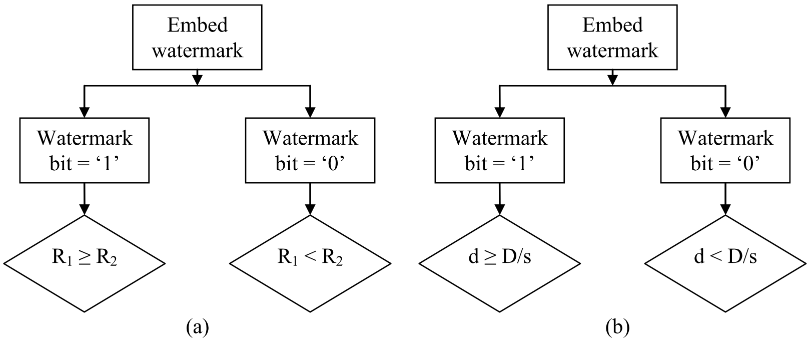


Fig. 6. Watermarking embedding algorithm: (a) Scheme 2 [31], (b) proposed scheme.

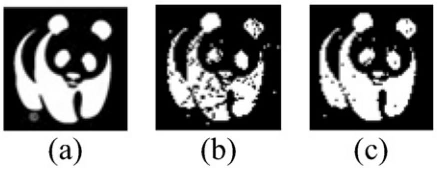


Fig. 7. Extracted watermark without tampering: (a) original 64 × 64 watermark, (b) using Scheme 2 [31], (c) using proposed scheme.

4.2. Proposed CRT algorithm

In watermarking Scheme 2 [31], the values of R_1 and R_2 were used in embedding watermark as shown in Fig. 6a. Whereas, in our proposed scheme, we make use of the values d (5) and D (6), and a scale factor s, in the watermarking algorithm as shown in Fig. 6b. We conducted an experiment to compare the performance of the proposed scheme with that of Scheme 2 by embedding watermark bits in the DCT domain and extracting the watermark without tampering the watermarked image. As shown in Fig. 7, the Scheme 2 performed worse than the proposed scheme. The proposed algorithm produces a low TAF value of 0.51%, whereas, Scheme 2 resulted a TAF value of 3.47%. The TAF will be further elaborated later in Section 5.

The extracted watermark shown in Fig. 7c does not appear exactly as the original watermark. This is because the DCT conversion introduces rounding errors during computation and also in the inverse DCT conversion. Hence, this will cause the resulting watermark to look slightly tampered even though no tampering was intentionally introduced. We have chosen the scale factor, s to be 2 when the DC location is selected and s to be 4 when AC location is selected. This provides

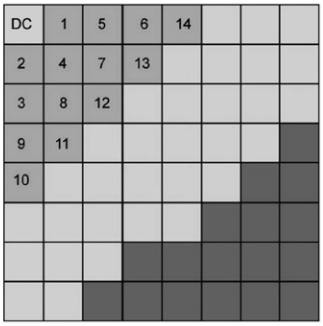


Fig. 8. Zigzag sequence of the 8 × 8 DCT block under JPEG.

sufficient embedding strength while not compromising on watermarked image quality for embedding in the respective DC and AC locations.

4.3. Selection of DCT coefficients

An important parameter of watermarking is to determine the embedding position for the watermark in the host image. For robustness, it is preferred to embed into the most significant component but this may degrade the image quality and the watermark may become visible. On the other hand, if perceptually insignificant components are used, watermark information may be lost during lossy compression. Since the proposed scheme uses a fixed block size of 8 × 8 pixels, the resulting DCT block is also of size 8 × 8 giving a total of 64 DCT coefficients as seen in Fig. 8. The DC coefficient and all the AC coefficients shown in a zigzag numbering sequence are used in JPEG compression. The DC coefficient represents the average value of the 64 image pixels and contains a significant portion of the total energy of the block of pixels. The remaining 63 DCT coefficients are AC coefficients which represent the frequency components of the block of pixels. In general, low frequency components contain more energy than high frequency components. Also, the human perception system is more sensitive to low frequency components. After JPEG compression, there is a high probability that the values of high frequency coefficients will be close to zero. Hence, from these characteristics of the coefficients, we decided that it is best to place emphasis on the DC coefficient and low frequency coefficients.

From experimental results that we have conducted, it would be better to include both DC [33] and low frequency AC [34] coefficients for embedding so that the trade-off between the quality and robustness of the watermarked image is optimal. As such, we have chosen DC and the first three low frequency AC coefficients as the possible locations for embedding the watermark bits in the DCT domain. With reference to Fig. 8, the first three AC coefficients refer to cells 1, 2 and 3, respectively. For choosing the AC coefficients, we took the AC coefficients in the order of the zigzag sequence as used in the JPEG compression technique.

4.4. Embedding procedure

The process begins by dividing the host image into 8 × 8 pixels blocks. The program will automatically select the appropriate number of watermark bits to be embedded into a block. For example, to embed a 32 × 32 binary watermark image into a 256 × 256 host image, the number of watermark bits per block would be 1 watermark bit per block. Therefore, there would be exactly one watermark bit embedded in each block. After dividing the host image into blocks, consider one block at a time to embed the watermark bits as given in the following steps:

1. Select a random 8 × 8 pixels block from the host image.
2. Apply DCT conversion to the selected 8 × 8 block.
3. Randomly select a watermark bit from the watermark image to embed into the block.
4. Randomly select a DCT coefficient (either the DC or one of the AC components) to embed the watermark bit. Let its value be denoted as Z.
5. Let M_1 and M_2 be the pair-wise co-prime numbers to be used in CRT with values 38 and 107, respectively, if Z is the DC coefficient. On the other hand, if Z is an AC coefficient, the values of M_1 and M_2 are selected as 38 and 55, respectively.
6. Applying the inverse CRT to Z, find R_1 and R_2.
7. Determine d using (5) and D using (6).
8. To embed watermark bit ‘1’, the required condition is

$$d \geqslant \frac{D}{s}, \tag{10}$$

where $s = 2$ if Z is DC coefficient, otherwise, $s = 4$. If (10) is not satisfied, then Z is modified to Z' until (10) is satisfied.

9. To embed watermark bit ‘0’, the required condition is

$$d < \frac{D}{s}. \tag{11}$$

If (11) is not satisfied, then Z is modified to Z' until (11) is satisfied. The process of embedding bit ‘0’ and bit ‘1’ is explained below.

10. Reconstruct the DCT block with the modified DCT coefficient, Z' and apply inverse DCT to the block to construct the watermarked image block.
11. Repeat steps 1–10 for the remaining blocks until all watermark information bits are embedded.

The following points should be considered before proposing the above scheme for embedding a watermark bit. The range of possible values for the DC and AC coefficients are not the same. For a given 8×8 pixels block where the pixel value is between 0 and 255; the value of the DC coefficient ranges from 0 to 2040, whereas, for the AC coefficients, it ranges from -1020 to $+1020$. According to CRT, the product of the pair-wise co-prime numbers (also called the dynamic range) should be greater than the largest possible number to be considered. In our case the largest possible value of the DC coefficient is 2040 and the product of the arbitrarily selected pair-wise co-prime number is 2047 (23×89). For AC coefficients, the largest absolute value is 1020 and the arbitrarily selected pair-wise co-prime numbers gives a product of 1035 (23×45). As such, we require two sets of values of co-prime numbers to be used when embedding and extracting. However, from experiments conducted, we have noted that by doubling the dynamic range, there would be better noise performance without much loss in image quality. The watermarked image quality with $2\times$ dynamic range was 41.42 dB compared to 45.49 dB for $1\times$ dynamic range. Hence, the values of the pair-wise co-prime numbers are changed to $\{38, 107\}$ (dynamic range $= 4066$) and $\{38, 55\}$ (dynamic range $= 2090$), respectively, for the DC and AC coefficients.

As we have mentioned, to embed watermark bits ‘1’ and ‘0’ there are some conditions to be satisfied. If those conditions are not fulfilled, there should be some modifications. Let us now consider the modifications done to satisfy the conditions. A flow-chart of the embedding procedure is also shown in Fig. 9.

1. *To embed bit ‘1’*: First check whether the condition (10) is satisfied. If it does, there is no need to modify the values of R_1 and R_2. If the condition does not satisfy, then add 8 to Z and continue from Step 6 of the embedding procedure with the new Z' and check whether (10) is satisfied for bit ‘1’ to be embedded. If adding 8 does not yield the result, try subtracting 8 from Z and continue with Step 6 of the embedding procedure and check whether (10) is satisfied. If subtracting 8 also does not yield the expected result; carry on adding and subtracting 8 to Z until (10) satisfies. After which the modified values of R_1 and R_2 would be used by CRT to get Z', the modified DCT coefficient.

2. *To embed bit ‘0’*: First check whether the condition (11) is satisfied. If it does, there is no need to modify the values of R_1 and R_2. If the condition does not satisfy, then add 8 to Z and continue from Step 6 of the embedding procedure with the new Z' and check whether (11) is satisfied for bit ‘0’ to be embedded. If adding 8 does not yield the result, try subtracting 8 from Z and continue with Step 6 of the embedding procedure and check whether (11) is satisfied. If subtracting 8 also does not yield the expected result; carry on adding and subtracting 8 to Z until (11) satisfies. After which the modified values of R_1 and R_2 would be used by CRT to get Z', the modified DCT coefficient.

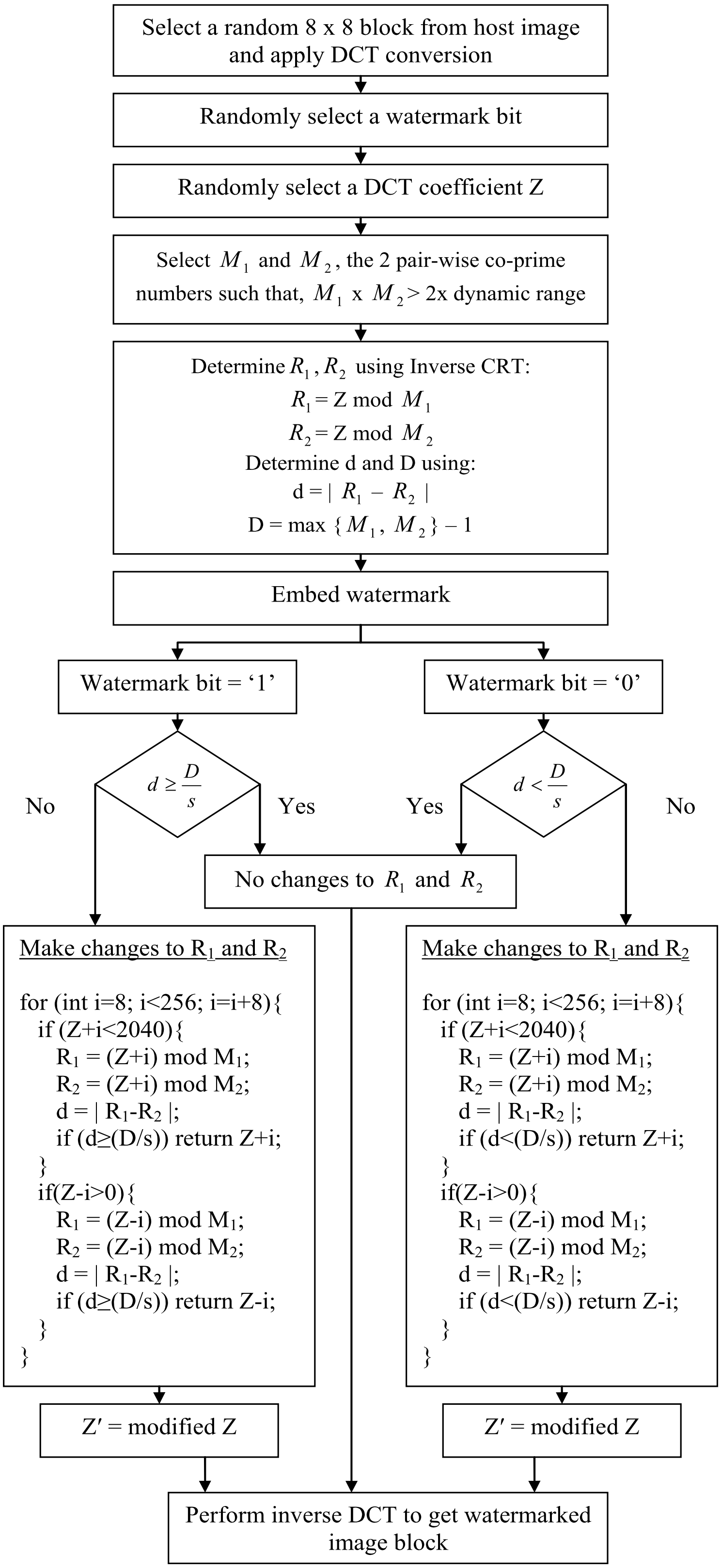
The reason for using ± 8 to make modifications to the selected DCT coefficient is that it provides sufficient amount of modification in the DCT domain that would be reflected back in the spatial domain. Assuming a watermark bit embedding into a DCT block with a change of $+1$ made to the DCT coefficient. When the DCT block undergoes inverse DCT to get the 8×8 pixels watermarked block, the resulting embedded block pixel values could be the same as before the watermark bit was embedded. Hence, this will result in “invalid” watermark information when we do the extraction. As such, we have done experiments and concluded that a change of 8 is sufficient for the watermark information to be maintained after undergoing DCT and inverse DCT conversion while minimizing distortion to the watermarked image. The maximum limit of change was set to 256 after testing all possible values of the DCT coefficient for embedding watermark bit one and zero.

4.5. Extraction procedure

The extraction procedure is reverse of the embedding procedure. We need to know only the following information to extract the watermark from the watermarked image:

1. Watermarked image.
2. Size of the watermark.
3. Seed of the PRNG (Pseudo Random Number Generator).
4. The pair-wise co-prime numbers M_1 and M_2.

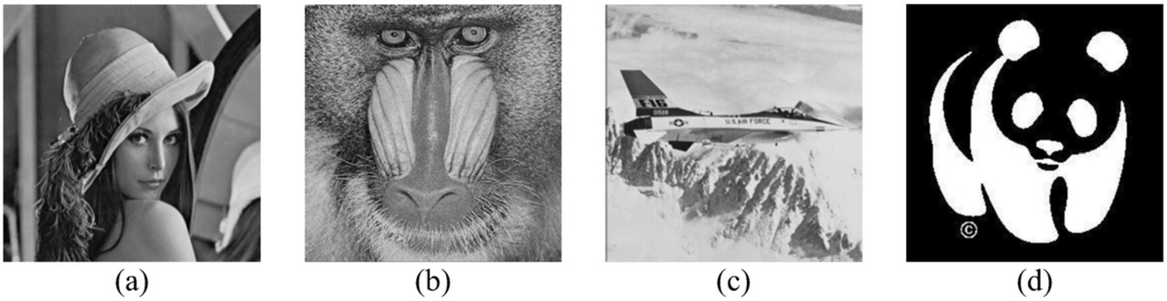
With the knowledge of PRNG, the DCT coefficient that is embedded with the watermark information Z is extracted. Thereafter, with the values of M_1, M_2 and Z, and using (4), R_1 and R_2 are determined, and d is determined using (5). Next, a comparison is made between d and D. If $d \geqslant \frac{D}{s}$, bit ‘1’ would be extracted, otherwise bit ‘0’ would be extracted. The scale factor, s is set to 2 for DC coefficient otherwise it is set to 4. These steps are repeated for every consecutive block to extract all the watermark bits.


Select a random 8 x 8 block from host image and apply DCT conversion
Randomly select a watermark bit
Randomly select a DCT coefficient Z
Select M_1 and M_2, the 2 pair-wise co-prime numbers such that, M_1 x M_2 > 2x dynamic range
Determine R_1, R_2 using Inverse CRT:
R_1 = Z mod M_1
R_2 = Z mod M_2
Determine d and D using:
d = | $R_1 - R_2$ |
D = max { M_1, M_2 } – 1
Embed watermark
Watermark bit = '1'
Watermark bit = '0'
$d \geq \frac{D}{s}$
$d < \frac{D}{s}$
No
Yes
Yes
No
No changes to R_1 and R_2
Make changes to R_1 and R_2
Make changes to R_1 and R_2
Z′ = modified Z
Z′ = modified Z
Perform inverse DCT to get watermarked image block

```
for (int i=8; i<256; i=i+8){
  if (Z+i<2040){
    R1 = (Z+i) mod M1;
    R2 = (Z+i) mod M2;
    d = | R1-R2 |;
    if (d≥(D/s)) return Z+i;
  }
  if(Z-i>0){
    R1 = (Z-i) mod M1;
    R2 = (Z-i) mod M2;
    d = | R1-R2 |;
    if (d≥(D/s)) return Z-i;
  }
}
```

```
for (int i=8; i<256; i=i+8){
  if (Z+i<2040){
    R1 = (Z+i) mod M1;
    R2 = (Z+i) mod M2;
    d = | R1-R2 |;
    if (d<(D/s)) return Z+i;
  }
  if(Z-i>0){
    R1 = (Z-i) mod M1;
    R2 = (Z-i) mod M2;
    d = | R1-R2 |;
    if (d<(D/s)) return Z-i;
  }
}
```

Fig. 9. Flow chart to embed a watermark bit.



(a) (b) (c) (d)

Fig. 10. The three host images (256 × 256): (a) Lenna, (b) Baboon, (c) Airplane, and (d) the watermark image, Panda (64 × 64).

5. Experimental results

In order to carry out the performance comparison of the proposed scheme with other schemes, two performance measures are defined below. The extent of tampering of the extracted watermark is computed using a Tamper Assessment Function (TAF) [31]. Considering the size of the watermark as $m \times n$, the TAF in percentage is defined as

$$\mathrm{TAF}(\%) = \frac{1}{mn}\left[\sum_{i=1}^{m}\sum_{j=1}^{n} w(i,j) \oplus \overline{w}(i,j)\right] \times 100, \tag{12}$$

where $w(i,j)$ and $\overline{w}(i,j)$ represent the original and extracted watermarks at position, (i,j) respectively, and $\oplus$ is an exclusive-OR operator. The TAF represents the number of bits of the extracted watermark those are different from the original watermark, expressed in percent. Usually, the acceptance level of TAF is 15%, because the extracted watermark will not be recognizable above this value.

Peak Signal-to-Noise Ratio (PSNR) measures the quality between two images. Usually we would compare the modified signal against the original signal, i.e., in this case, the watermarked image against the original host image. The value of PSNR usually ranges from 20 dB (low quality) to 40 dB (high quality). Since the host images used in the experiments are in 8-bit grayscale format, the peak value of the image is taken to be 255. The PSNR in dB is given by

$$\mathrm{PSNR\ (dB)} = 10\log_{10}\left[\frac{255^2}{\frac{1}{M}\frac{1}{N}\sum_{i=1}^{M}\sum_{j=1}^{N}[A(i,j) - \overline{A}(i,j)]^2}\right], \tag{13}$$

where A represents the host image, $\overline{A}$ represents the watermarked image, and $M \times N$ represent the size of the two images.

In this study we carried out performance comparison among the three schemes considering the following performance criteria:

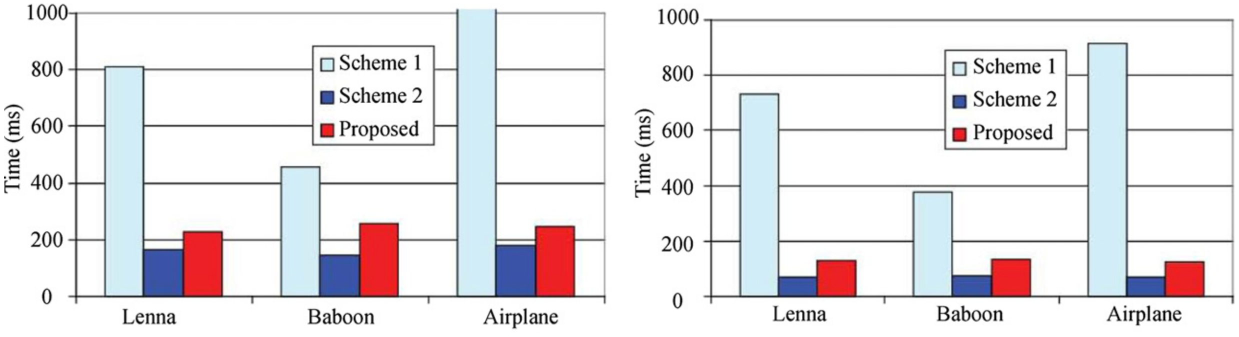
1. Computational complexity of the embedding and extracting procedures.
2. Quality of the watermarked images.
3. Robustness of the schemes to different attacks.
4. Further increase in embedding capacity.

5.1. Computational complexity

The computational time for embedding and extraction is an important performance measure, especially when the watermarking is to be carried out for online applications, e.g., video or audio broadcasting. Several experiments were conducted to evaluate performance of the proposed scheme against the other two schemes. The experiments were carried out in a laptop with Intel Core 2 Duo, 2.40 GHz clock and 4 GB RAM. Three benchmark graylevel images of 512 × 512 pixels of Lenna, Baboon and Airplane were used as host images, as shown in Fig. 10. A black and white (binary) image of Panda (Fig. 10d) was used as the watermark. Several watermark dimensions were used in the simulations, i.e., 64 × 64 and 128 × 64. The results are based on Lenna (host image, 512 × 512) watermarked with Panda (watermark, 64 × 64), unless otherwise stated. A strength factor, $\beta = 0.02$ was used in Scheme 1 [21]. We have observed that the results of simulations using the other two host images (Baboon and Airplane) were similar to that of Lenna.

We compare the embedding and extraction time of the proposed scheme against Scheme 1 [21] and Scheme 2 [31]. Ten simulation runs were conducted for each scheme to determine the average timing for each scheme and are plotted in Fig. 11. The host images were watermarked with Panda image (64 × 64).

From Fig. 11, one can see that the proposed scheme is able to embed and extract the watermark in much less time than Scheme 1 [21], but the proposed scheme takes slightly more time than Scheme 2 [31]. On averaging over the three host images, the embedding time for Scheme 1, Scheme 2 and proposed scheme were found to be 766, 164 and 245 ms, respectively. The proposed scheme is faster than Scheme 1 because only simple CRT calculations are required for embedding. In



(a)

(b)

Fig. 11. Comparison of embedding and extracting times for Scheme 1 [21], Scheme 2 [31] and proposed scheme: (a) embedding time, (b) extraction time.

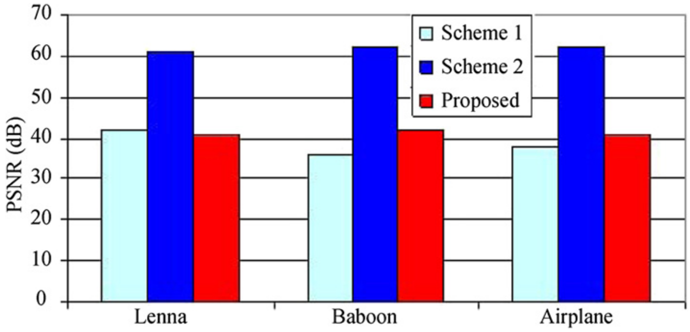


Fig. 12. Quality of watermarked image in PSNR for different host images in Scheme 1 [21], Scheme 2 [31] and proposed scheme.

the proposed scheme, two computational steps, i.e., the random selection of watermark blocks and bits, and the conversion to and from DCT domain makes up the bulk of the embedding time. These two additional computational steps explain why the proposed scheme is slower than Scheme 2 even though both are CRT-based. Scheme 1 takes a significant amount of time due to computation of SVD, calculation of ranks of the individual blocks and selection of blocks to embed the watermark bits. Furthermore, it can also be seen that the time taken to select blocks based on rank is image-dependent that gives rise to significant differences in timing for Scheme 1. However, in our proposed scheme, the embedding time is found to be independent of the host image. The average extraction times for the Scheme 1, Scheme 2 and proposed scheme were observed to be 676, 73 and 129 ms, respectively. It was also noticed that unlike the SVD-based Scheme 1, the extraction time in the proposed scheme does not depend on the host image.

5.2. *Quality of watermarked images*

The quality of the watermarked images with respect to the different schemes is investigated here. Fig. 12 shows the quality of the watermarked images based on PSNR for the different host images. In Scheme 1, a strength factor, $\beta = 0.02$ was used for embedding. It can be seen that the proposed scheme maintains a similar if not higher quality watermarked image than the SVD-based Scheme 1 for different host images. Also, it can be noted that both Scheme 2 and proposed scheme have consistent watermarked image quality compared to the SVD-based scheme which seems to be image dependent. The PSNR of the watermarked image in the proposed scheme is about 42 dB for the Lenna host image.

5.3. *Robustness against attacks*

In this section, we compare the proposed scheme against Scheme 1 [21] and Scheme 2 [31] in their ability to withstand different types of attacks. The watermarked image was subjected to different attacks, e.g., addition of noise and tampering, before extracting the watermark. The quality of the extracted watermark is determined by their TAF value (12). A lower TAF value would indicate that the extracted watermark is more similar to the original watermark.

Several image manipulation techniques were used to distort the watermarked images [7–9,31]. These manipulations are: (i) cropping of a block size of 10% in the middle of the watermarked image; (ii) tampering 10% in the middle of the watermarked image on 10% of the pixels with a strength factor of 30; (iii) adding noise to the entire watermarked image with a 25% distortion rate; (iv) brightening the watermarked image to 110%; (v) sharpening the watermarked image by 50%; and (vi) JPEG compression with compression level of 0.75. Samples of the watermarked images under attacks stated above and using Lenna as the host image are shown in Fig. 13.

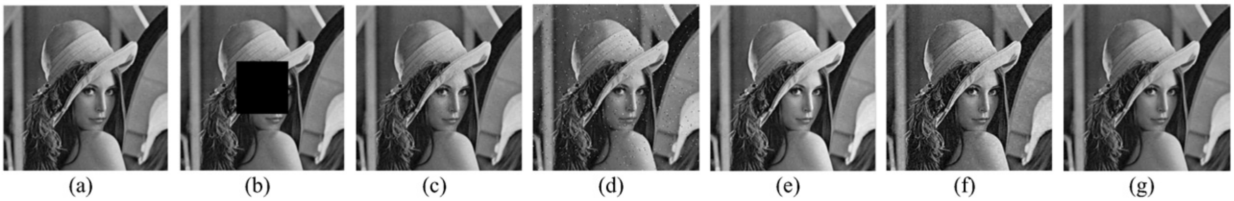


Fig. 13. Watermarked Lenna image under different attacks: (a) without attack, (b) crop, (c) tamper, (d) addition of noise, (e) brightening, (f) sharpening, (g) JPEG compression.

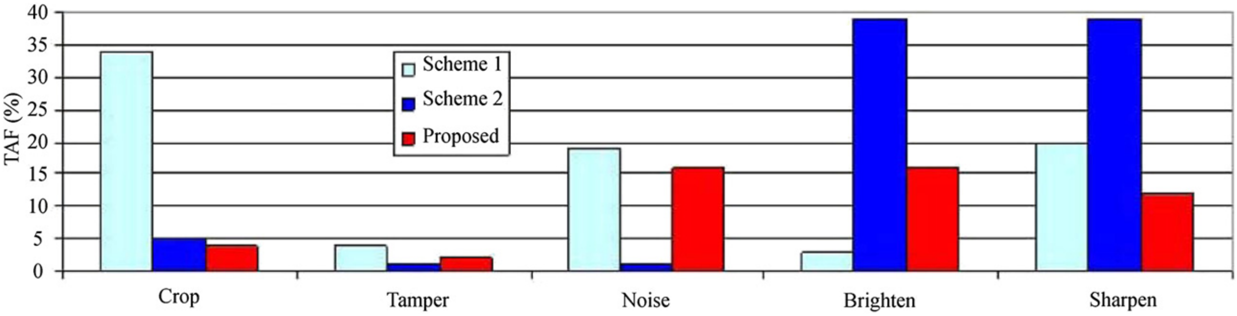


Fig. 14. Quality of extracted watermark after cropping and tampering attacks: Scheme 1 [21], Scheme 2 [31] and proposed scheme.

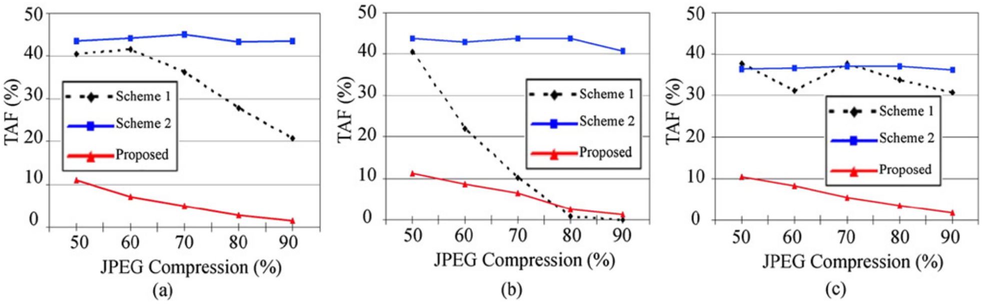


Fig. 15. TAF of extracted watermark under JPEG attack in Scheme 1 [21], Scheme 2 [31] and proposed scheme: (a) Lenna, (b) Baboon, (c) Airplane.

Fig. 14 shows the TAF of the extracted watermarks after the watermarked images were subjected to crop, tamper, noise, brighten and sharpen attacks, respectively. The results are based on host image Lenna (512×512), watermark image Panda (64×64) and block size $= 8 \times 8$. A strength factor, $\beta = 0.02$ was used in Scheme 1 [21]. For cropping and tampering attacks, the area affected was the centre of the watermarked image. For both attacks, the TAF performance of the proposed scheme is similar to that of Scheme 2 [31] but much superior than the SVD-based Scheme 1.

For the noise attack, the watermarked image PSNR is reduced to 25 dB by introducing additive white Gaussian noise to the entire watermarked image. As seen from Fig. 14, the proposed scheme does not fare as well as Scheme 2, however, it performs better than Scheme 1. This is due to the fact that the noise is spread out in the DCT domain among the coefficients and hence affects the embedded watermark information. It can be seen from Fig. 14 that there is substantial improvement in brightness and sharpening performance over Scheme 2 as the effects of brightening and sharpening the pixel values are uniformly spread out in the whole DCT block.

The watermarked image was compressed by JPEG algorithm with varying levels ranging from 50 to 90%. Three host images, Lenna, Baboon and Airplane of size 512×512 pixels and the watermark Panda of size 64×64 pixels were used in this test. As seen from Fig. 15, the proposed scheme is quite robust to varying levels of JPEG compression: the TAF value remains under 10% for JPEG compression level from 50% to 90%. In contrast, Scheme 2 [31] is not able to withstand any JPEG compression since it produces quite high TAF value which causes the extracted watermark unrecognizable. The performance of Scheme 1 [21] is dependent on the host image used, but its TAF value is much higher than the proposed scheme. For all three test images used, the proposed scheme maintained the lowest TAF value.

The experimental results for the JPEG compression, brightening and sharpening attacks showed that the proposed scheme is more robust compared to Scheme 2. For cropping and tampering attacks, the proposed scheme performed similar to Scheme 2 and better than Scheme 1. However, the proposed scheme does not fare well under severe noise attacks. All these tests were conducted on the other two host images, i.e., Baboon and Airplane, and similar results were observed.

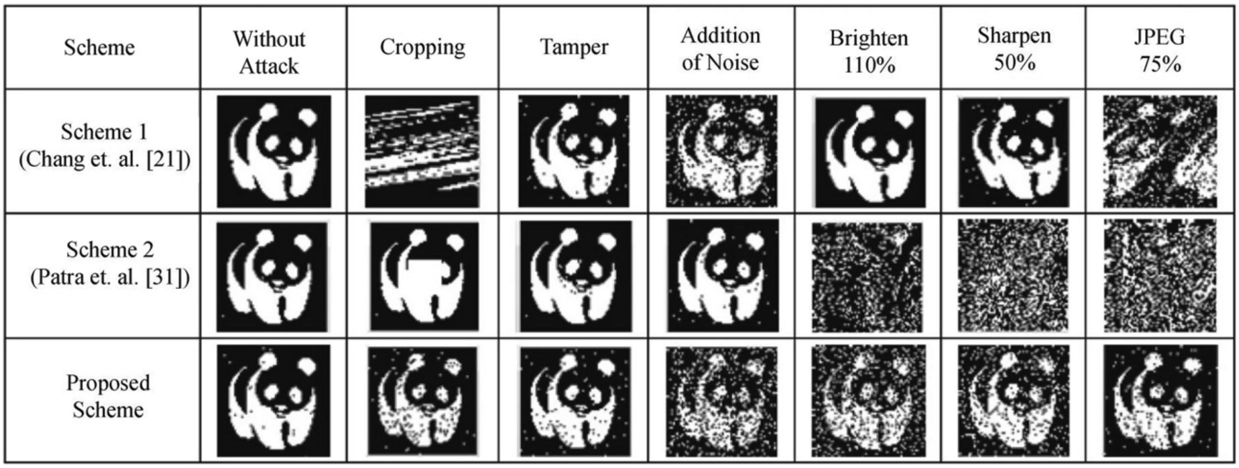


Fig. 16. Extracted watermarks from the attacked watermarked Lenna image (watermark size = 64 × 64).

Table 1
PSNR and TAF obtained under various attacks using: (a) Scheme 1 [21], (b) Scheme 2 [31], (c) proposed scheme.

Image	Quality of the embedded image and extracted watermark under different attacks														
	Scheme	(a) Without attack		(b) Cropping		(c) Tampering		(d) Addition of noise		(e) Brightening		(f) Sharpening		(g) JPEG (75%)	
		PSNR	TAF	PSNR	TAF	PSNR	TAF	PSNR	TAF	PSNR	TAF	PSNR	TAF	PSNR	TAF
Lenna	1	42.47	2.73	16.80	33.62	37.25	3.81	24.96	19.04	27.38	**2.73**	25.58	19.85	38.90	39.16
	2	**60.97**	**0.05**	**16.81**	5.30	**38.77**	**0.71**	**25.02**	**0.81**	**27.53**	38.65	**27.12**	39.09	**41.41**	44.29
	Proposed	*41.42*	*0.51*	*16.79*	***4.20***	*36.90*	*2.17*	*24.92*	*16.53*	*27.30*	*15.82*	*26.43*	***11.55***	*38.46*	***4.91***
Baboon	1	36.29	0.00	13.98	33.47	34.28	0.42	24.74	7.64	25.32	**0.00**	14.28	**5.57**	30.09	**2.59**
	2	**61.52**	**0.00**	**14.01**	5.27	**38.56**	**0.32**	**25.01**	**0.88**	**25.70**	43.07	**14.40**	45.34	**31.28**	43.87
	Proposed	*41.93*	*0.68*	*14.00*	***4.20***	*36.93*	*2.32*	*24.97*	*18.65*	*25.57*	*19.36*	*14.37*	*31.91*	*30.92*	*5.13*
Airplane	1	38.23	0.49	13.37	33.84	35.39	28.30	24.83	28.42	22.80	39.11	21.65	35.40	36.92	34.42
	2	**61.75**	**0.00**	13.38	5.27	**38.57**	**0.49**	**25.00**	**0.39**	**22.94**	35.62	**23.39**	35.30	**51.89**	37.43
	Proposed	*41.40*	*0.22*	***13.38***	***4.00***	*36.75*	*2.39*	*24.92*	*16.33*	*22.84*	***19.29***	*23.08*	***15.70***	*41.04*	***6.05***

Some extracted watermark images for different attacks are shown in Fig. 16. It can be seen that for the proposed scheme, there is a significant improvement in the quality of extracted watermark from the watermarked image subjected to JPEG compression. There is also noticeable improvement for brightening and sharpening attack compared to Scheme 2. For cropping attack, the proposed scheme performed better when compared to Schemes 1 and 2. The use of random location of embedding the watermark bits is highlighted here as we compare to Scheme 2 where the cropped middle region of the watermarked image results in a "white box" in the middle of the extracted watermark. For tampering attack, the proposed scheme has similar performance compare to Schemes 1 and 2. However, the proposed scheme is not as robust as Scheme 2, against noise attacks. The probable reason for the unsatisfactory performance is that since the noise spreads over all the DCT coefficients randomly, it adversely affects the extraction process.

The watermarked image quality (PSNR) and TAF values of the extracted watermarks under different attacks are summarized in Table 1. It can be seen that the proposed scheme outperforms other two schemes against JPEG compression attack. We also note that Scheme 1's performance is image dependent while Scheme 2 and the proposed scheme's performance are not image dependent, especially in brightening and sharpening attacks. For cropping, tampering and addition of noise, the proposed scheme provides consistent performance over the three host images.

5.4. *Increase of watermark capacity*

Here, we compare the performance of the proposed scheme only with Scheme 2, because there is no provision of increase of watermark capacity in the SVD-based Scheme 1 [21]. The proposed scheme utilizes all the blocks in the host image to embed the watermark bits. The watermark embedding capacity in the host image can be doubled by embedding two bits per block instead of one. This is done by embedding the second bit randomly by selecting another DCT coefficient from the remaining three available DCT coefficients in the watermarking block. We have seen that by doing so, the quality of the watermarked image (PSNR) drops about 3 dB when all other conditions remain the same. This drop in image quality is not as significant as differences in watermarked images with increased embedded bits per block are not noticeable to the

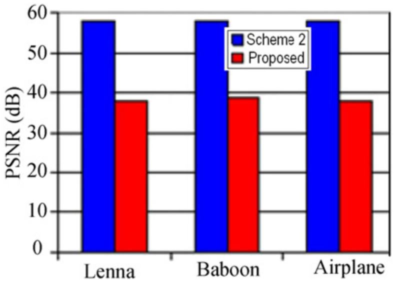


Fig. 17. Quality of watermarked image (PSNR) for different host images after doubling of watermarking capacity in Scheme 2 [31] and the proposed scheme.

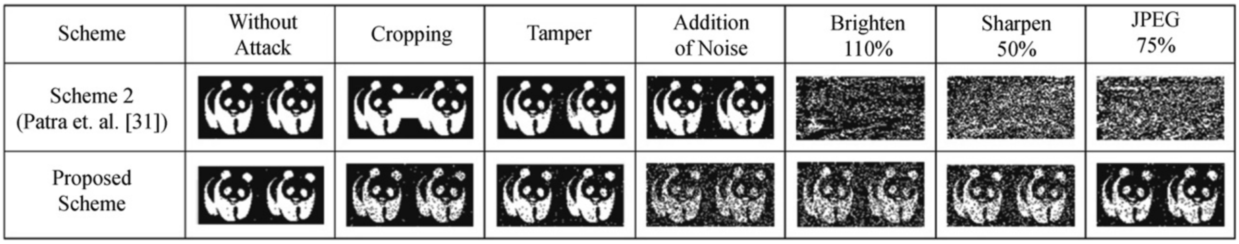


Fig. 18. Extracted watermarks from the attacked watermarked Lenna image using Scheme 2 [31] and proposed scheme (watermark size = 128 × 64).

human eye. Fig. 17 shows the results of the proposed scheme based on host image Lenna (512 × 512), watermark image Panda (128 × 64) and block size = 8 × 8. The watermarked image is still considered to be of high quality (≈ 40 dB). Fig. 18 shows the quality of extracted watermarks under different attacks. Despite doubling of watermark capacity, the proposed scheme maintains its performance similar to that of one watermark bit embedded per block (see Fig. 16). In case of JPEG compression, the proposed scheme maintains its superiority over Scheme 2.

6. Conclusion

In this paper we have proposed a new approach based on CRT for watermarking of images in the DCT domain for authentication and copyright protection. The use of CRT provides advantage in terms of improved security and low computational complexity. In addition, the proposed scheme features added security due to its random selection of watermarking blocks and random selection of location of watermark bit to embed. We have compared its performance with a SVD-based and a spatial domain CRT-based watermarking scheme, and shown that the proposed scheme outperforms the other two under some major image attacks, such as, brightening and sharpening. Especially, the proposed scheme exhibits strong resistance to the JPEG compression. Since CRT involves only modular operations for its computation, the time required for embedding and extraction of watermark in the proposed scheme is much less compared to the SVD-based scheme. We have shown that even after doubling the watermark capacity, the proposed scheme maintains its imperceptibility and robustness against various attacks. However, the proposed scheme is not robust against severe noise attacks. Currently we are working to solve this problem. The proposed scheme introduces an effective and efficient watermarking technique for images which may be equally applicable to other forms of digital media, e.g., text, audio and video.

References

[1] W. Bender, D. Gruhl, N. Morimoto, A. Lu, Techniques for data hiding, IBM Systems J. 35 (3&4) (1996) 313–336.
[2] I.J. Cox, J. Kilian, F.T. Leighton, T. Shamoon, Secure spread spectrum watermarking for multimedia, IEEE Trans. Image Process. 6 (12) (Dec. 1997) 1673–1687.
[3] I.J. Cox, M.L. Miller, A.L. Mckellips, Watermarking as communication with side information, Proc. IEEE 87 (7) (Jul. 1999) 1127–1141.
[4] N. Nikolaidis, I. Pitas, Robust image watermarking in the spatial domain, Signal Process. 66 (Sep. 1997) 385–403.
[5] P.W. Wong, N. Memon, Secret and public key image watermarking schemes for image authentication and ownership verification, IEEE Trans. Image Process. 10 (10) (Oct. 2001) 1593–1601.
[6] T.H. Chen, D.S. Tsai, Owner–customer right protection mechanism using a watermarking scheme and a watermarking protocol, Pattern Recogn. 39 (8) (Aug. 2006) 1530–1541.
[7] C.Y. Lin, S.F. Chang, A robust image authentication method distinguishing JPEG compression from malicious manipulation, IEEE Trans. Circuits Systems Video Technol. 11 (2) (Feb. 2001) 153–168.
[8] C.T. Hsu, J.L. Wu, Hidden digital watermarks in images, IEEE Trans. Image Process. 8 (1) (Jan. 1999) 58–68.
[9] M. Barni, F. Bartolini, V. Cappellini, A. Piva, A DCT-domain system for robust image watermarking, Signal Process. 66 (Nov. 1997) 357–372.
[10] H. Qi, D. Zheng, J. Zhao, Human visual system based adaptive digital image watermarking, Signal Process. 88 (Jul. 2007) 174–188.
[11] K.-C. Liu, Wavelet-based watermarking for color images through visual masking, AEU Int. J. Electronics Commun. 64 (2) (Feb. 2010) 112–124.
[12] T.V. Nguyen, J.C. Patra, A simple ICA-based watermarking scheme, Digital Signal Process. 18 (2008) 762–776.
[13] Y. Xin, S. Liao, M. Pawlak, Circularly orthogonal moments for geometrically robust image watermarking, Pattern Recogn. 40 (May 2007) 3740–3752.

[14] A.M. Ahmed, D.D. Day, Applications of the naturalness preserving transform to image watermarking and data hiding, Digital Signal Process. 14 (2004) 531–549.
[15] C. Deng, X. Gao, X. Li, D. Tao, A local Tchebichef moments-based robust image watermarking, Signal Process. 89 (2009) 1531–1539.
[16] R. Liu, T. Tan, An SVD-based watermarking scheme for protecting rightful ownership, IEEE Trans. Multimedia 4 (1) (Mar. 2002) 121–128.
[17] A.A. Mohammad, A. Alhaj, S. Shaltaf, An improved SVD-based watermarking scheme for protecting rightful ownership, Signal Process. 88 (2008) 2158–2180.
[18] J.M. Shieh, D.C. Lou, M.C. Chang, A semi-blind digital watermarking scheme based on singular value decomposition, Comput. Stand. Interf. 28 (4) (Apr. 2006) 428–440.
[19] S. Lee, D. Jang, C.D. Yoo, An SVD-based watermarking method for image content authentication with improved security, in: Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Process. vol. 2, March 2005, pp. 525–528.
[20] K.L. Chung, W.N. Yang, Y.H. Huang, S.T. Wu, Y.C. Hsu, On SVD-based watermarking algorithm, Appl. Math. Comput. 188 (1) (2007) 54–57.
[21] C.-C. Chang, P. Tsai, C.-C. Lin, SVD-based digital image watermarking scheme, Pattern Recogn. Lett. 26 (2005) 1577–1586.
[22] J.C. Patra, W. Soh, E.L. Ang, P.K. Meher, An improved SVD-based watermarking technique for image and document authentication, in: Proc. IEEE Asia Pacific Conference on Circuits and Systems, Singapore, December 2006, pp. 1984–1987.
[23] E. Kougianos, S.P. Mohanty, R.N. Mahapatra, Hardware assisted watermarking for multimedia, Comp. Electrical Eng. 35 (2009) 339–358.
[24] Y.H. Ku, X. Sun, The Chinese remainder theorem, J. Franklin Inst. 329 (I) (1992) 93–97.
[25] T.-S. Chen, Y.-F. Chung, Hierarchical access control based on Chinese remainder theorem, Comp. Security 2 (6) (2002) 565–570.
[26] C.-C. Chang, Y.-P. Hsieh, C.-H. Lin, Sharing secrets in stego images with authentication, Pattern Recogn. 41 (2008) 3130–3137.
[27] S. Myung, K. Yang, A combining method of quasi-cyclic LDPC codes by Chinese remainder theorem, IEEE Commun. Lett. 9 (9) (Sep. 2005) 823–825.
[28] E. Al-Radadi, P. Siy, RNS sign detector based on Chinese remainder theorem II (CRT II), Comp. Math. Appl. 46 (2003) 1559–1570.
[29] C.-C. Chang, J.-S. Lee, Robust t-out-of-n oblivious transfer mechanism based on CRT, J. Network Comp. Appl. 32 (1) (Jan. 2009) 226–235.
[30] J.C. Patra, A. Karthik, P.K. Meher, C. Bornand, Robust CRT-based watermarking technique for authentication of image and document, in: Proc. IEEE Int. Conf. Systems, Man, and Cybernetics (SMC 2008), Singapore, October 2008, pp. 3250–3255.
[31] J.C. Patra, A. Karthik, C. Bornand, A novel CRT-based watermarking technique for authentication of multimedia contents, Digital Signal Process. 20 (2010) 442–453.
[32] Bruce Schneier, Applied Cryptography, 2nd ed., John Wiley & Sons, New York, 1996.
[33] J. Huang, Y.Q. Shi, Y. Shi, Embedding image watermarks in DC components, IEEE Trans. Circuits Systems Video Technol. 10 (6) (Sep. 2000) 974–979.
[34] Y. Wang, A. Pearmain, Blind MPEG-2 video watermarking robust against geometric attacks: A set of approaches in DCT domain, IEEE Trans. Image Process. 15 (June 2006) 1536–1543.

Jagdish C. Patra received B.Sc. (Engg.) and M.Sc. (Engg.) degrees, both in electronics & telecommunication engineering, from Sambalpur University, India, in 1978 and 1989, respectively. He obtained Ph.D. degree in electronics & communication engineering from Indian Institute of Technology, Kharagpur, India, in 1997. After receiving Bachelor's degree, he served in various R&D, teaching and government organizations for about eight years. In 1987, he joined Regional Engineering College, Rourkela, India, as a Lecturer, where he was promoted to Assistant Professor in 1990. In April 1999, he visited Technical University, Delft, Netherlands, as a Guest Teacher (Gastdocent), for six months.

Since 2001, he is serving as an Assistant Professor in the School of Computer Engineering, Nanyang Technological University, Singapore. His research interests include intelligent information processing using neural networks in the areas of data security, sensor networks, image processing, modeling of solar cells and bioinformatics. In these areas, he has over 100 research publications in international journals of repute and top international conferences. He is a Member of IEEE (USA) and Life Member of Institution of Engineers (India).

Jiliang E. Phua received Bachelor of Engineering (Computer Engineering) from Nanyang Technological University, Singapore, in 2010. After receiving the Bachelor's degree, he is currently working at Institute for Infocomm Research, Singapore, in the Networking Protocols Department. His research interests include image processing, biomedical signal processing using wavelets, noninvasive health care monitoring using optical systems and optical sensor networks.

Cédric Bornand graduated from the Swiss Federal Institute of Technology, Lausanne (EPFL), in 1988. Currently, he is a full professor at the University of Applied Sciences, HEIG-VD, Switzerland, and manages a research group in the field of embedded sensors. Over the last twenty years his team has carried out research in various aspects of sensors applications, ranging from mechanical measurements to traffic regulation, developing effective online algorithms as well as hardware implementations. Cédric Bornand focused his activities on multimodal systems and biomedical applications, with a strong effort in the integration of algorithms on various DSP solutions.